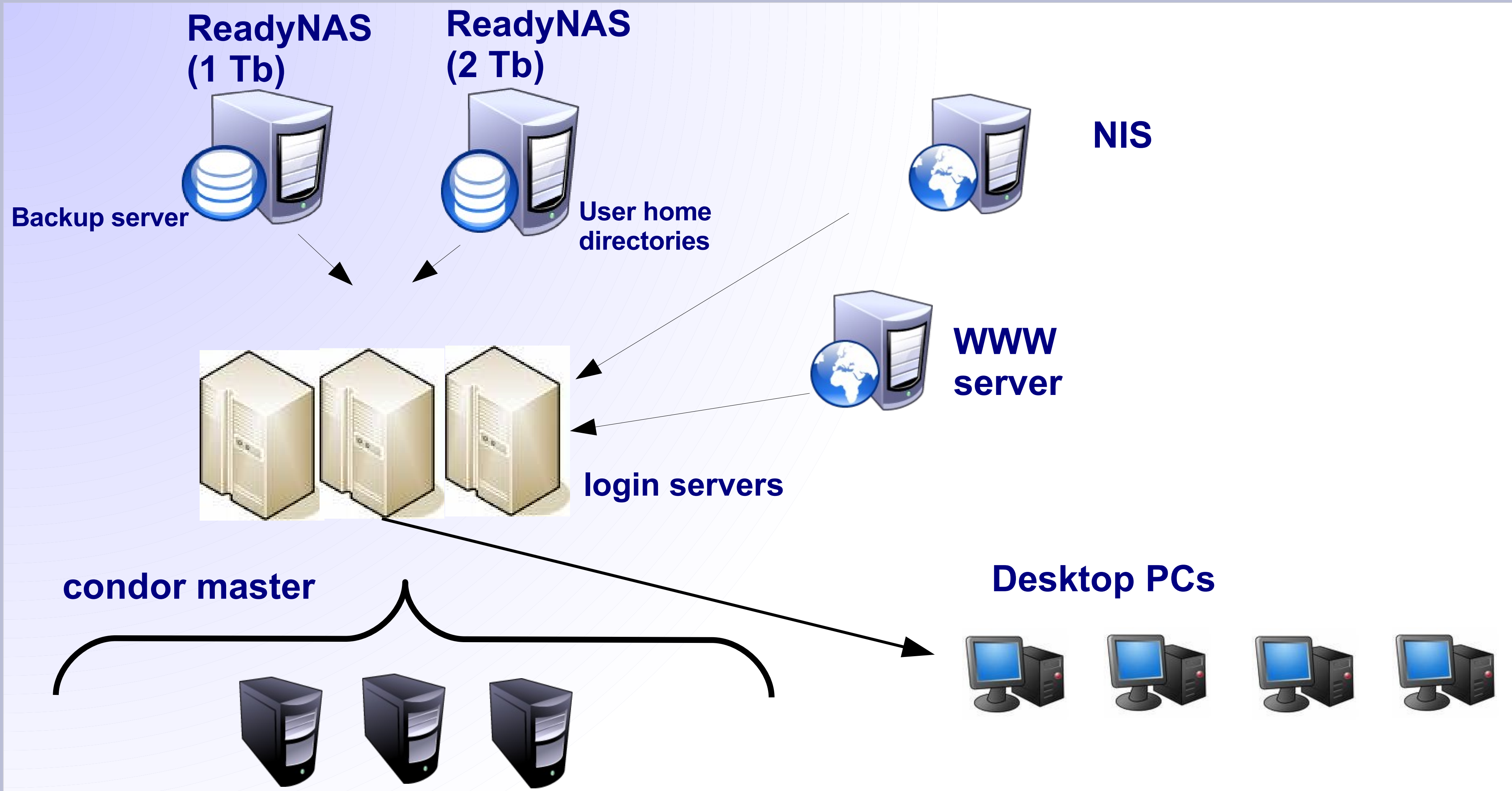
ANL PC farm tutorial and the ArCond job submission

S.Chekanov (ASC-ANL)

http://atlaswww.hep.anl.gov/asc/

Present infrastructure



ReadyNAS (1 Tb)

Backup server

ReadyNAS (2 Tb)

User home directories

NIS

WWW server

login servers

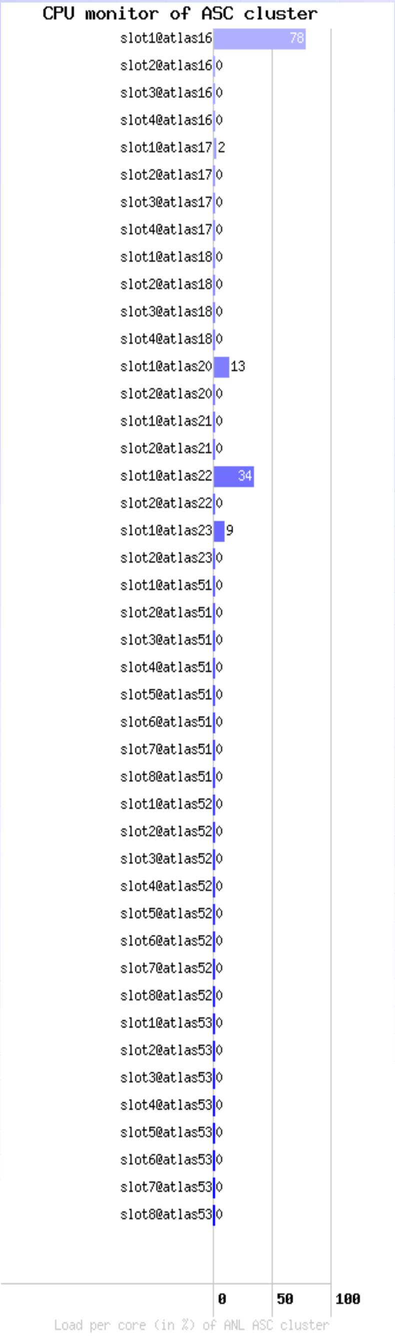
condor master

Desktop PCs

3 boxes, 24 cores,   6 TB space

Xeon 2.33 GHz processor:

Initial PC farm in action



http://atlaswww.hep.anl.gov/asc/admin/cpu-monitor/

public ssh login servers

Desktop PCs

PC farm prototype

Do you really need this?

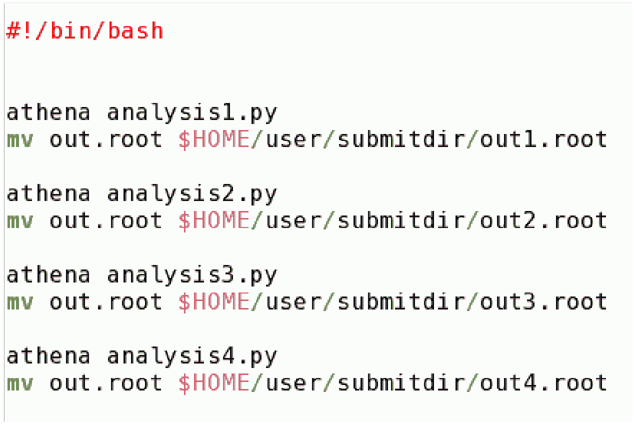
Yes, for post-Grid analyses (processing skimmed APD's, DPD's, user ntuples)

Obvious advantage:

- Jobs can be run 30 - 40 times faster compare to 4- CPU PCs (like atlas16-18)
- No any load on NFS
- Plenty of file storage (6 Tb)

Some advantages compare to the grid:

- 0% rate for unsuccessful jobs (for the grid runs ~ 10-20%)
  - statistics from several hundreds of runs
- No need the grid certificate (good for students!)
- Can run ANY program including bash shell script
  - you still cannot do this using pathena”?
- Do not want to recompile the program on the farm?
  - Send InstallArea!
- Can be easily debugged
  - Just login on faulty PC node and check & rerun
- I can help to customize submission scrips



```
#!/bin/bash


athena analysis1.py
mv out.root $HOME/user/submitdir/out1.root

athena analysis2.py
mv out.root $HOME/user/submitdir/out2.root

athena analysis3.py
mv out.root $HOME/user/submitdir/out3.root

athena analysis4.py
mv out.root $HOME/user/submitdir/out4.root
```

ArCond

- Use ArCond (Argonne Condor) for submission
- Similar to PROOF, but based on Condor
- Allows to submit any job, identify data on the PC farm boxes,
- Checks job status, merges outputs etc.

How to start?

- Login to atlas16-18
- Get the package:

```
cp /users/condor/Arcond/ .
```

- Go to the Arcond directory:

```
cd Arcond
export PATH=$PATH:`pwd`
```

Setup atlas release as usual (run this script)

```
export AVERS=14.2.21                        # ATLAS release
export TEST_AREA=$HOME/testarea   # testarea location
source /share/grid/app/asc_app/asc_rel/1.0/setup-script/set_atlas.sh
```

You are ready!

Simple example

- Let us run a script “example.sh” on each PC boxes (not on each CPU!)

```
#! /bin/sh

echo "HELLO, ArCond!"
echo "HOSTNAME is: " `hostname`
echo "The job starts at: " `date`

echo "Users online: "  `users`
echo "CPU load: "  `uptime`


echo "Show loaded data on local storage:"
ls -lt /data1/mc/PythiaZeegam25/aod
```

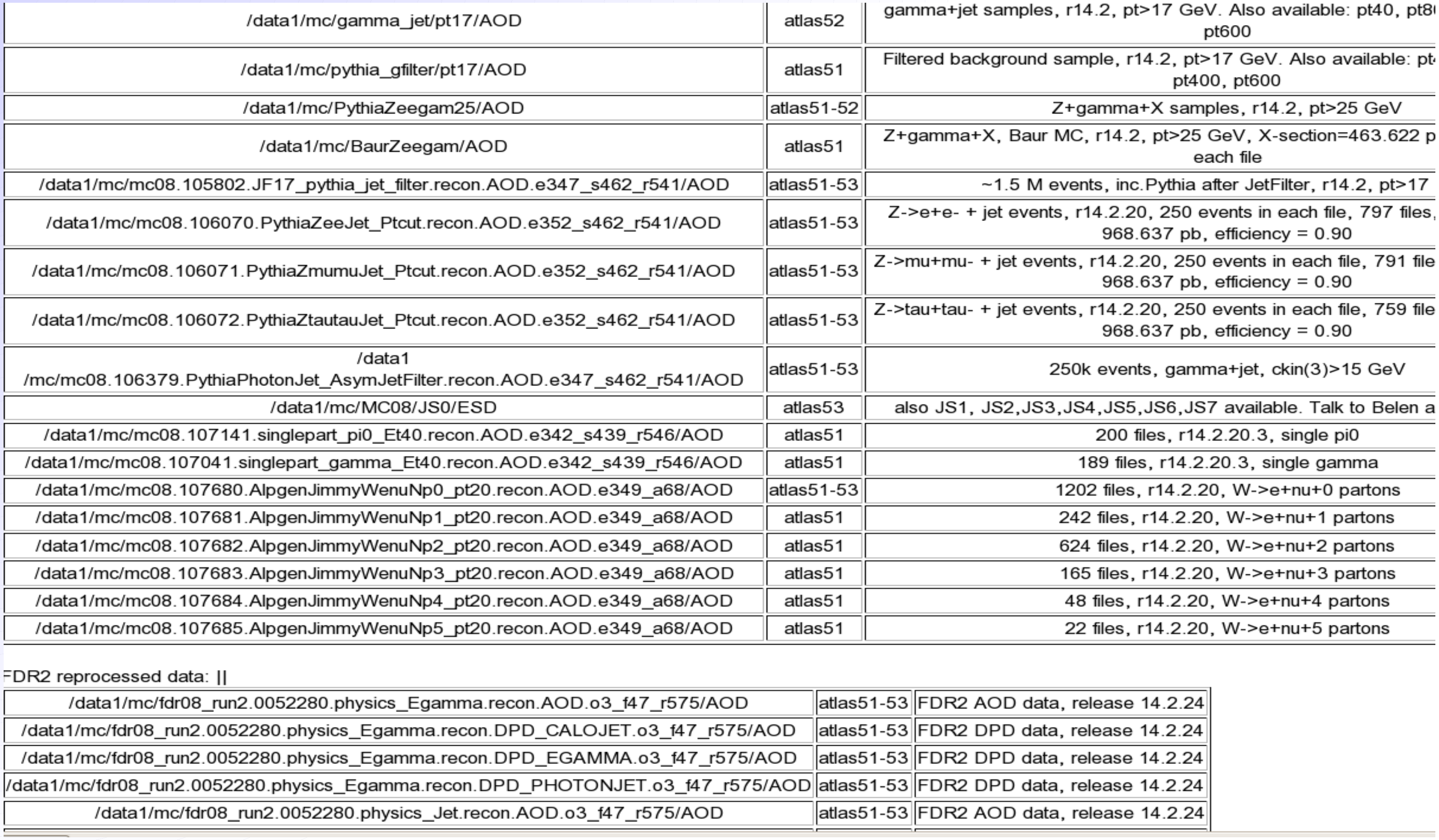
Execute the line: arc_exe -i example.sh

arcond will print the output of this script on each PC box (takes 3-10 sec)

Sending Athena jobs

Before sending an athena job, make sure that the data set is available on the PC farm. Check:

http://atlaswww.hep.anl.gov/asc/ASC_working/index.php?n=Main.DataSamples



/data1/mc/gamma_jet/pt17/AOD	atlas52	gamma+jet samples, r14.2, pt>17 GeV. Also available: pt40, pt8 pt600
/data1/mc/pythia_gfilter/pt17/AOD	atlas51	Filtered background sample, r14.2, pt>17 GeV. Also available: pt pt400, pt600
/data1/mc/PythiaZeegam25/AOD	atlas51-52	Z+gamma+X samples, r14.2, pt>25 GeV
/data1/mc/BaurZeegam/AOD	atlas51	Z+gamma+X, Baur MC, r14.2, pt>25 GeV, X-section=463.622 p each file
/data1/mc/mc08.105802.JF17_pythia_jet_filter.recon.AOD.e347_s462_r541/AOD	atlas51-53	~1.5 M events, inc.Pythia after JetFilter, r14.2, pt>17
/data1/mc/mc08.106070.PythiaZeeJet_Ptcut.recon.AOD.e352_s462_r541/AOD	atlas51-53	Z->e+e- + jet events, r14.2.20, 250 events in each file, 797 files, 968.637 pb, efficiency = 0.90
/data1/mc/mc08.106071.PythiaZmumuJet_Ptcut.recon.AOD.e352_s462_r541/AOD	atlas51-53	Z->mu+mu- + jet events, r14.2.20, 250 events in each file, 791 file 968.637 pb, efficiency = 0.90
/data1/mc/mc08.106072.PythiaZtautauJet_Ptcut.recon.AOD.e352_s462_r541/AOD	atlas51-53	Z->tau+tau- + jet events, r14.2.20, 250 events in each file, 759 file 968.637 pb, efficiency = 0.90
/data1 /mc/mc08.106379.PythiaPhotonJet_AsymJetFilter.recon.AOD.e347_s462_r541/AOD	atlas51-53	250k events, gamma+jet, ckin(3)>15 GeV
/data1/mc/MC08/JS0/ESD	atlas53	also JS1, JS2,JS3,JS4,JS5,JS6,JS7 available. Talk to Belen a
/data1/mc/mc08.107141.singlepart_pi0_Et40.recon.AOD.e342_s439_r546/AOD	atlas51	200 files, r14.2.20.3, single pi0
/data1/mc/mc08.107041.singlepart_gamma_Et40.recon.AOD.e342_s439_r546/AOD	atlas51	189 files, r14.2.20.3, single gamma
/data1/mc/mc08.107680.AlpgenJimmyWenuNp0_pt20.recon.AOD.e349_a68/AOD	atlas51-53	1202 files, r14.2.20, W->e+nu+0 partons
/data1/mc/mc08.107681.AlpgenJimmyWenuNp1_pt20.recon.AOD.e349_a68/AOD	atlas51	242 files, r14.2.20, W->e+nu+1 partons
/data1/mc/mc08.107682.AlpgenJimmyWenuNp2_pt20.recon.AOD.e349_a68/AOD	atlas51	624 files, r14.2.20, W->e+nu+2 partons
/data1/mc/mc08.107683.AlpgenJimmyWenuNp3_pt20.recon.AOD.e349_a68/AOD	atlas51	165 files, r14.2.20, W->e+nu+3 partons
/data1/mc/mc08.107684.AlpgenJimmyWenuNp4_pt20.recon.AOD.e349_a68/AOD	atlas51	48 files, r14.2.20, W->e+nu+4 partons
/data1/mc/mc08.107685.AlpgenJimmyWenuNp5_pt20.recon.AOD.e349_a68/AOD	atlas51	22 files, r14.2.20, W->e+nu+5 partons

FDR2 reprocessed data: ||

/data1/mc/fdr08_run2.0052280.physics_Egamma.recon.AOD.o3_f47_r575/AOD	atlas51-53	FDR2 AOD data, release 14.2.24
/data1/mc/fdr08_run2.0052280.physics_Egamma.recon.DPD_CALOJET.o3_f47_r575/AOD	atlas51-53	FDR2 DPD data, release 14.2.24
/data1/mc/fdr08_run2.0052280.physics_Egamma.recon.DPD_EGAMMA.o3_f47_r575/AOD	atlas51-53	FDR2 DPD data, release 14.2.24
/data1/mc/fdr08_run2.0052280.physics_Egamma.recon.DPD_PHOTONJET.o3_f47_r575/AOD	atlas51-53	FDR2 DPD data, release 14.2.24
/data1/mc/fdr08_run2.0052280.physics_Jet.recon.AOD.o3_f47_r575/AOD	atlas51-53	FDR2 AOD data, release 14.2.24

Short rules:

- **Data set can be located on any PC box.**
- **Use unique path if data are distributed among several PC.**
- **Always make “info” directory with log files**
- **Duplicate files will be excluded automatically by arcond**

S.Chekanov: ANL ASC Tier3 infrastructure

Running arcond

- Before submitting a job, prepare a configuration file (“ arcond.conf”)

```
atlas_release=14.2.21

# events to process in each job
events = -1

# dir with input AOD files.
# input_data = /data1/mc/mc08.105802.JF17_pythia_jet_filter.recon.AOD.e347_s462_r541/AOD

# package directory on NFS
package_dir = /users/chakanau/testarea/14.2.21/analysis/PromptGamma
```

scan all subdirectories!

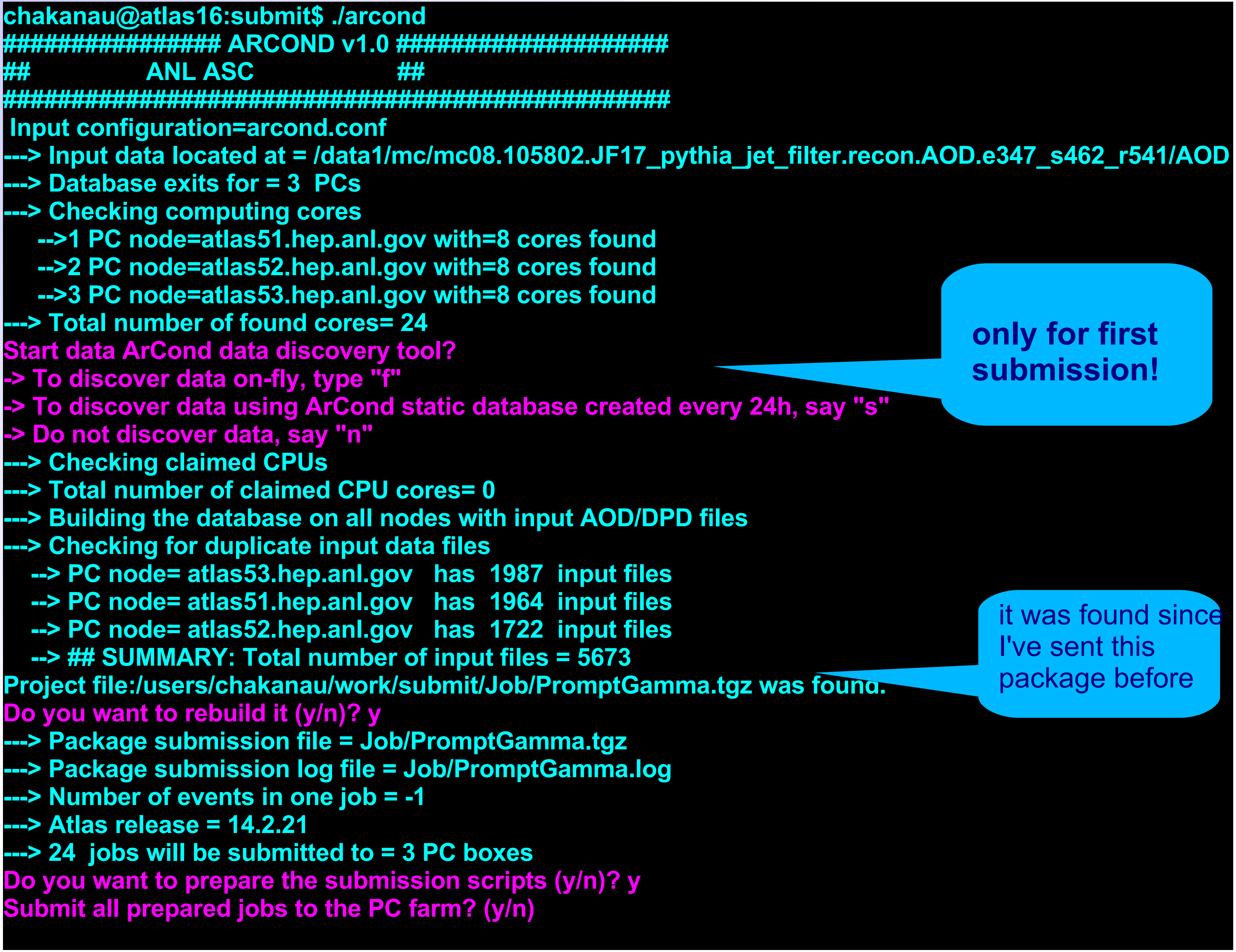
- Prepare the job option file you want to run:

- Copy your job option file to: Arcond/user/Analysis_jobOptions_BASIC.py
  - 1) Remove the line "theApp.EvtMax" (defined from config file)
  - 2) Better use “Analysis.root” as output
  - 2) Append the line:

```
include("InputCollection.py")
ServiceMgr.EventSelector.InputCollections = dataCollection
```

Ready to submit!

Submitting job..



```
chakanau@atlas16:submit$ ./arcond
################# ARCOND v1.0 #####################
##              ANL ASC                  ##
##################################################
 Input configuration=arcond.conf
---> Input data located at = /data1/mc/mc08.105802.JF17_pythia_jet_filter.recon.AOD.e347_s462_r541/AOD
---> Database exits for = 3  PCs
---> Checking computing cores
    -->1 PC node=atlas51.hep.anl.gov with=8 cores found
    -->2 PC node=atlas52.hep.anl.gov with=8 cores found
    -->3 PC node=atlas53.hep.anl.gov with=8 cores found
---> Total number of found cores= 24
Start data ArCond data discovery tool?
-> To discover data on-fly, type "f"
-> To discover data using ArCond static database created every 24h, say "s"
-> Do not discover data, say "n"
---> Checking claimed CPUs
---> Total number of claimed CPU cores= 0
---> Building the database on all nodes with input AOD/DPD files
---> Checking for duplicate input data files
   --> PC node= atlas53.hep.anl.gov   has  1987  input files
   --> PC node= atlas51.hep.anl.gov   has  1964  input files
   --> PC node= atlas52.hep.anl.gov   has  1722  input files
   --> ## SUMMARY: Total number of input files = 5673
Project file:/users/chakanau/work/submit/Job/PromptGamma.tgz was found.
Do you want to rebuild it (y/n)? y
---> Package submission file = Job/PromptGamma.tgz
---> Package submission log file = Job/PromptGamma.log
---> Number of events in one job = -1
---> Atlas release = 14.2.21
---> 24  jobs will be submitted to = 3 PC boxes
Do you want to prepare the submission scripts (y/n)? y
Submit all prepared jobs to the PC farm? (y/n)
```

only for first submission!

it was found since I've sent this package before

Arcond tips

- Tired to say “yes”? Run ArCond in quite mode “arcond -allyes”
- Check all options as “arcond -h”
- If you want to repeat submission:
  - do not rebuild database in: “/DataCollector” directory
  - do not need to rebuild submission package in “/Job” directory
- If the PC farm is busy (Check as: condor_q -global):
  - you cannot rebuild the database (timeout error)
  - you can still submit jobs (but they will be put to “idle” state)
- After the submission, you will see submission directories in “/Job”
  - Do not modify anything there!

Checking and getting jobs back

- Run condor commands: condor_status or condor_q
- Pay attention to atlas51-53 nodes
- Your jobs are in “idle” state?
  - check who is running on the farm as:
    - condor_status -submitters (OR) condor_q -global
- Check output files as: arc_check
- If arc_check tells that all output files “Analysis.root” are ready, combine output files to one file using arc_add. This creates “Analysis_all.root”

- To debug program and check errors:
  - ./Job/runXXX/Analysis.log - athena log file
  - ./Job/runN_atlasXXX/Job.ShellScript.atlasXXX/job.local.out - execution log

ArCond for IT experts

- Pure python & bash. Does not need installation. Requires condor.
  - > arcond
    - Reads a configuration file (with atlas release version, input directory with AOD files on all boxes, package athena name)
    - Splits jobs to be run in parallel: N=N(PC boxes) x N(cores)
    - Data discovery using local storage. Builds a database with input files and associates each AOD file with specific box
    - Splits data lists, prepare submission SHELL scripts, submits to each box with local data
      - shell submission script contains anything you like, including multiple athena runs
    - Compiles the job using either NFS-based ATLAS software release or locally installed release
    - Runs jobs using local condor home directory
    - When jobs are ready, the output is copied to submission directory
    - Output root files merged automatically
    - Automatic check new arcond version

ArCond for advanced users

- You can exclude any PC box from the submission by removing files in "patterns/ schema.site.atlasXX.cmd"

- ArCond can run any custom sequence determined in a submission shell

- The submission shell is in *user/ShellScript_BASIC.sh*

- Do not modify XXvariableXX

- The most important statements in the script (for athena):

  - cmt config; source setup.sh; make;
  - athena Analysis.py

- Can also run any C++/ROOT compiled program
  - i.e. put: "make clean; make; main.exe; etc"

- Can put any debugging bash statement

- Compilation statements can be removed if you tar InstallArea and copy to the farm

Summary

- PC farm prototype is functional
- Several people including myself are running only on the PC farm
- Since Sep 1, ~3000 jobs completed on each CPU node
  - ~ 150 completed jobs (24 – 8 CPs used for each run)
- Example for 24 cores:
  - to process 8.5M AOD events (JF17) takes ~ 20h
    - ~560 h on a single CPU of atlas16
  - to process ntuples created from the same sample takes 4 min
    - ~ 2h on a single CPU of atals16
- So far no any failures
  - some problems may appear if the condor master is busy
  - will be replaced soon with a dedicated condor master PC
- More nodes will be added soon, extending the farm to ~100 nodes
- Try it!